



Monitorprogramm für den MC9S12XD256

Sebastian Sterk und Manuel Weyreter

Elektro-/Informationstechnik plus

WS 2008 / 2009

Inhaltsverzeichnis

1	Einleitung	1
2	Funktionen	1
2.1	Kommunikation	1
2.2	Run/Load Mode	1
2.3	Schreibzugriff auf Flash und EEPROM	2
2.4	Memory Map	2
2.5	Monitorbefehle	3
2.5.1	Blank check	3
2.5.2	Dump Memory	3
2.5.3	Edit Memory	3
2.5.4	Fill Memory	4
2.5.5	Goto	4
2.5.6	Help	4
2.6	System Info	4
2.6.1	Load	4
2.6.2	Move Memory	4
2.6.3	Select PPage	5
2.6.4	Erase Flash	5
2.6.5	Erase EEPROM	5

1 Einleitung

Dieses Dokument gibt einen Überblick über die Funktionen und Befehle des „TwinPEEKs“ Monitorprogramms von Oliver Thamm, der das Programm zur Verwendung im Labor Embedded Systems der Hochschule Offenburg zur Verfügung gestellt hat.

2 Funktionen

2.1 Kommunikation

„TwinPEEKs“ kommuniziert über „SCI0“ des S12XD mit 19200 Baud.

Weitere Einstellungen: 8 Datenbits, 1 Stoppbit, kein Paritätsbit, kein Hardware- oder Software-Handshake und kein Protokoll.

2.2 Run-mode und load-mode

Der „TwinPEEKs Monitor“ überprüft nach einem Reset, ob Port E Pin 2 (PTE2) auf Masse liegt. Ist dies der Fall, springt der Monitor zur Adresse „\$4000“. Hierdurch wird es möglich, ein Anwenderprogramm automatisch zu starten, ohne den Resetvektor im geschützten Flash Block ändern zu müssen.

Falls PTE2 auf „high“ (5 Volt) liegt, wird der Monitor normal gestartet.

Zur Anzeige das der Monitor ausgeführt wird, kann eine LED an Port E Pin 3 (low aktiv) angeschlossen werden.

2.3 Schreibzugriff auf Flash und EEPROM

Die CPU kann auf alle Ressourcen des Mikrocontrollers, unabhängig vom Speichertyp, byteweise lesend zugreifen.

Das Flash und das EEPROM müssen vor der Programmierung gelöscht werden.

Da die Programmierung wortweise erfolgt und der Zugriff stets auf eine gerade Wortadresse stattfinden muss, folgt daraus, dass zwei aufeinander folgende Einzelbytes zunächst zu einem, auf eine Wortgrenze ausgerichteten Wort, zusammengefasst werden.

„TwinPEEKs“ verarbeitet „S19-Daten“ stets zeilenweise. Falls nun die letzte belegte Adresse in einer solchen „S19-Zeile“ gerade ist, fehlt zunächst das für die Wort-Programmierung erforderliche zweite Byte. Somit wird ein „\$FF-Byte“ ergänzt und das Datenwort programmiert. Setzt sich der Datenstrom in der folgenden „S19-Zeile“ mit dem zuvor fehlenden Byte fort, müsste „TwinPEEKs“ an der fraglichen Wortadresse einen erneuten Schreibzugriff vornehmen, dies ist jedoch unzulässig. Somit kommt es zu einem Schreibfehler („not erased“). Daher ist es notwendig „S-Record-Daten“ vor der Programmierung auf gerade Adressen auszurichten.

Der Assembler, der in die „MiniIDE“ Entwicklungsumgebung integriert ist, erstellt schon entsprechend aufgebaute „S19-Dateien“.

2.4 Memory Map

Die Memory Map des Controllers wird von „TwinPEEKs“ wie folgt initialisiert:

von	bis	Belegung
\$0000	\$07FF	Register
\$0800	\$0FFF	EEPROM
\$1000	\$3FFF	RAM
\$4000	\$7FFF	Flash: Entspricht Page FD
\$8000	\$BFFF	Flash Pages E0...E7 und F8...FF mittels PPage wählbar
\$C000	\$DFFF	Flash
\$F000	\$FFFF	Geschützter Bereich

2.5 Monitorbefehle

Ein Monitorbefehl besteht aus einem Buchstaben, ggf. gefolgt von einem oder mehreren Argumenten und wird mit der „Enter-Taste“ abgeschlossen. Alle Zahlenangaben erfolgen hexadezimal ohne weitere Vor- oder Nachsätze. Außerdem wird nicht zwischen Groß- und Kleinschreibung unterschieden. Da der für die CPU sichtbare Adressraum 64KB umfasst, sind Adressargumente maximal vierstellig. Die Endadressen beziehen sich stets auf das dem Adressbereich folgende Byte. Der Befehl „D 1000 1200“ zeigt so z.B. den Adressbereich von „\$1000“ bis inklusiv „\$11FF“ an.

Mit dem Monitorprompt wird die aktuell verwendete Page ausgegeben.

In der folgenden Übersicht sind benötigte Argumente durch eckige Klammern und optionale Argumente durch geschweifte Klammern gekennzeichnet.

2.5.1 Blank check

Syntax: B

Prüft, ob der gesamte Flash Speicher (exkl. Monitorbereich) gelöscht ist. Falls dies nicht der Fall ist, wird die Nummer der ersten Page ausgegeben, in der ein Byte ungleich „\$FF“ gefunden wurde.

2.5.2 Dump Memory

Zeigt den Speicherinhalt ab Adresse „adr1“ bis Adresse „adr2“. Ohne Angabe einer Endadresse werden die folgenden „\$40“ Bytes angezeigt. Der Inhalt von „adr1“ wird durch eine eckige Klammer hervorgehoben.

2.5.3 Edit Memory

Editiert den Speicher. Nach der Startadresse „addr“ können bis zu vier Bytes angegeben werden. Die Daten werden unmittelbar geschrieben, danach kehrt die Funktion zur Eingabeaufforderung zurück.

Sind keine Daten in der Eingabezeile angegeben, wird der interaktive Modus gestartet. Dort wird der Speicher angezeigt und kann direkt verändert werden.

„Enter“ nächste Adresse „-“ vorhergehende Adresse

„=“ gleiche Adresse „.“ oder „Q“ Ende

2.5.4 Fill Memory

Syntax: F [adr1 adr2 byte]

Füllt den Speicherbereich ab Adresse „adr1“ bis exklusive „adr2“ mit dem Wert „byte“.

2.5.5 Goto

Syntax: G [addr]

Ruft das Anwenderprogramm ab Adresse „addr“ auf. Ein Rücksprung zum Monitor ist nicht vorgesehen.

2.5.6 Help

Syntax: H

Listet eine Kurzübersicht zu allen Monitorkommandos auf.

2.6 System Info

Syntax: I

Zeigt die Start- und Endadressen von Registerblock, RAM, EEPROM und Flash des Mikrocontrollers an und gibt die „PARTID“ des Mikrocontrollers aus.

2.6.1 Load

Syntax: L

Lädt eine „.S19-Datei“ in den Speicher. Es werden Zeilen vom Typ „S1“ (16-Bit MCU-Adressen) und „S2“ (lineare 24-Bit Adressen) verarbeitet. „S0“ Zeilen (Kommentarzeilen) werden übersprungen. „S8-“ bzw. „S9-Zeilen“ werden als „End-of-File-Markierung“ erkannt. Weiter Informationen sind in Kapitel 2.3 zu finden.

2.6.2 Move Memory

Syntax: M [adr1 adr2 adr3]

Kopiert den Speicherbereich ab Adresse „addr1“ bis exklusive Adresse „addr2“ nach Adresse „addr3“ und folgende.

2.6.3 Select PPage

Wählt eine Flash Page (PPAGE) aus. Diese Page wird daraufhin im 16KB-Page-Window von „\$8000“ bis „\$BFFF“ sichtbar.

Achtung: Die erlaubten Pages liegen im Bereich von „E0“ bis „E7“ und „F8“ bis „FF“. Der Speicherbereich von „\$4000“ bis „\$7FFF“ entspricht dem Inhalt der Page „FD“.

Ohne Angabe eines Arguments wird die aktuell verwendete Page ausgegeben.

2.6.4 Erase Flash

Löscht die angegebene Page (16KB) des Flashspeichers.

Ohne Angabe von „page“ löscht der Befehl den gesamten Flash, abgesehen vom Monitorcode.

2.6.5 Erase EEProm

Syntax: Y

Löscht das gesamte EEProm.